Claims

1. A method of processing data in a distributed computing environment wherein a client and a server process data, the method comprising sending the server from a first place where it communicates with the client, through the distributed computing environment towards a second different place to perform data processing therefrom.

2. A method according to claim 1 including freezing incoming calls for data processing to the server at the first place whilst it is being sent from the first place to the second place, and thereafter directing the frozen calls towards the second place to be processed by the server when it has become functional at the second place.

3. A method according to claim 2 including waiting for the server to complete its current processing tasks before sending it to the second place.

a

4. A method according to Claim 1 ~~any preceding claim~~ including converting the server from an operational configuration at the first place into a configuration suitable for transmission through the distributed environment to the second place.

5. A method according to claim 4 wherein the conversion comprises serialisation of the server.

a

6. A method according to Claim 1 ~~any preceding claim~~ including creating a proxy for the server at the first place, which controls the sending of the server towards the second place.

a

7. A method according to Claim 1 ~~any preceding claim~~ including sending the client towards a different place in the distributed computing environment.

8. A method of processing data in a distributed computing environment wherein a client and a server process data, the method comprising receiving the server sent from a first place where it communicated with the client, through the distributed computing environment, at a second different place, to perform data processing at the second place.

9. A method according to claim 8 wherein the server is received at the second place in a form suitable for transmission through the distributed environment, and including converting the received server at the second place into a form suitable for processing data at the second place.

10. A method according to claim 9 wherein the converting includes deserialising the server.

a 11. A method according to claim 8, ~~9 or 10~~ including producing a proxy for the received server, at the second place.

a 12. A method according to Claim 8 ~~any one of claims 8 to 11~~ including receiving at the second place, data processing calls for the server directed thereto from the first place after the server has become operational at the second place.

13. A software entity operable to provide a server for a client in a distributed computing environment **characterised in that** the software entity is selectively re-locatable to different places through the environment.

14. An entity according to claim 13, operable to function as the server at a first place in the environment and then to re-locate and function as the server at a second place in the environment.

a 15. An entity according to claim 13 ~~or 14~~, operable such that data calls thereto from a client are frozen during the re-location.

16. An entity according to claim 13 ~~any one of claims 13 to 15~~ operable to provide a proxy functional to send the server through the environment to achieve the re-location.

17. An entity according to claim 16 wherein the proxy is functional to wait for the server to complete its current processing tasks before commencing the re-location.

18. An entity according to claim 16 ~~or 17~~ wherein the proxy is operable to serialise the server from its functional configuration into a configuration suitable for transmission through the distributed environment so as to achieve the re-location.

19. A software entity according to claim 13 ~~any one of claims 13 to 18~~, stored on a storage medium.

20. A signal for transmission in a distributed computing environment wherein a client and a server process data, the signal comprising the server serialised for transmission between a first place where it communicates with the client, through the distributed computing environment and a second different place to perform data processing.

21. A proxy for use in a distributed computing environment wherein a client and a server process data, the proxy being operable to send the server from a first place where it communicates with the client, through the distributed computing environment towards a second different place to perform data processing.

22. A proxy according to claim 21 operable to freeze incoming calls for data processing to the agent at the first place whilst it is being sent from the first place to the second place, and thereafter to direct the frozen calls towards the second place to be processed by the server when it has become functional at the second

place.

23. A proxy according to claim 21 ~~or 22~~ operable to wait for the server to complete its current processing tasks before sending it to the second place.

24. A proxy according to claim 21, ~~22 or 23~~ operable to serialise the server from an operational configuration at the first place into a configuration suitable for transmission through the distributed environment to the second place.

25. A host provided with client and server objects for processing data in an object oriented distributed processing environment **characterised in that** the server object is selectively re-locatable to different places in the environment.

26. A host according to claim 25 wherein the mobile server object is operable such that data calls thereto are frozen during the relocation.

27. A host according to claim 25 wherein the server is provided with a proxy compatible with CORBA or OLE architecture.

28. A server object for processing data in an object oriented distributed processing environment **characterised in that** the server object is re-locatable for operation at different places and is provided in use with a proxy which freezes data calls thereto during the relocation and subsequently forwards them to the moved server object.

add B' ?